APPARATUS AND METHOD

FOR MANAGING PERSISTENT

NETWORK CONNECTIONS

Michael T. Frantzen

David E. Ballman

William R. Danielson

FIELD

[0001] The present invention generally relates to techniques for establishing and maintaining network connections. The present invention also relates to techniques for providing network security.

BACKGROUND

[0002] A company's assets are at risk when it connects to the Internet. Unrestricted access and sharing of data and other resources may create serious security problems. For example, it is highly desirable to protect certain sensitive data from outside intruders, while making these data freely available to company employees accessing it from within the company's own network. In the recent years, a number of techniques have been developed to protect corporate private network against unauthorized use and to generally control access thereto. One of the most common techniques for securing a private network is the use of a firewall. A firewall is a highly secure host that acts as a barrier between internal network, such as a private corporate network, and all outside networks, such as the Internet. A firewall has two functions. Firstly, it acts as a gateway which passes data between the networks. Secondly, it acts as a barrier that blocks free passage of data to and from the private network. More specifically, the firewall computer is configured such that it allows network connections that are permitted by the company's security policy and refuses all the others.

[0003] The most commonly utilized type of firewall architecture is a packet–filtering firewall. It is well known in the art that most modern network communication devices communicate using data packets. For example, a TCP/IP packet contains, among other data,

2

information on the network address and the connection port of the sender, information on the

network address and the connection port of the recipient, and information on the type of the

communication protocol used. The firewall uses the aforementioned information to filter out the

packets of the network connections that are in violation of the security policy. For example, the

firewall may be configured to filter out all data packets sent from outside the private network,

except for the packets originating in specific hosts presumed to be secure and specified by the

security policy of the network.

[0004] One specific type of packet filtering firewall architecture is a stateful firewall.

Once any specific network connection is established across the firewall, the stateful firewall

stores the state of each such network connection in its database. The network connection state

entry includes, among other data, the network address and port information of the sender, the

network address and port information of the recipient and the time of the last packet transfer.

Each data packet corresponding to any specific network connection is handled by the stateful

firewall in accordance with a state of this connection stored in the firewall's database. One

example of a stateful firewall is Sun Screen firewall developed by Sun Microsystems of Palo

Alto, California.

[0005] If a particular connection is not active for an extended period of time, a stateful

firewall will assume that the connection has expired and then it will delete the connection by

removing the connection state information entry from its database. This is done to prevent

unrecoverable memory consumption. This aspect of operation of the stateful firewall is

illustrated in Fig. 1. Specifically, the firewall checks at 10 whether the connection is idle. This

is done, for example, by computing the time interval since the last packet transfer corresponding

to this connection. If the connection is idle, the firewall simply deletes the connection state entry from the database at 11, which destroys the connection. The operation of the algorithm terminates at 12. If the firewall determines that the connection is not idle, is does not delete the connection state from its database.

[0006] On the other hand, it is desirable for some applications, such as telnet to allow very long periods of user's inactivity. Telnet is an application that communicates with a remote host using a TELNET protocol, enabling a user to execute shell commands on the remote host and displaying the output of these commands. Both the telnet command and the TELNET protocol are well known in the art. For example, a user may want to telnet into a host, perform some actions on that host, and leave the telnet idle for several days.

[0007] Then the user may want to continue using the same connection several days later. It would be convenient if the user would not have to re-authenticate himself. But in the above example, the conventional stateful firewall will have likely deleted the connection after a few hours of user's inactivity. Thus, the user returning to work days later will discover that his telnet connection has hung. Thus, the user will have use the telnet to establish a new connection to the remote host and authenticate himself again by entering his name and a secret password. This lengthy process would be unnecessary if the firewall would recognize persistent connections and keep them "alive" for extended periods of time.

SUMMARY

[0008] To overcome the limitations described above, and to overcome other limitations that will become apparent upon reading and understanding the present specification, apparatus, methods and articles of manufacture are disclosed that keep persistent connections alive in a

network configuration involving a stateful firewall.

[0009]   One aspect of the invention is a method for managing a network connection in a network configuration comprising a firewall.

[0010]   Another aspect of the invention is a computer readable medium containing a program for managing network connections is a network architecture including a firewall.

[0011]   Yet another aspect of the invention is a firewall configured to manage network connections.

[0012]   According to the invention, the firewall automatically determines whether the network connection is active; and deletes a state of the network connection if the network connection is not active.

[0013]   The firewall may determine the condition of the network connection by generating a probe, which causes a network activity corresponding to the network connection in question. The firewall subsequently senses this network activity to determine whether the network connection is active.

[0014]   The firewall may include a database for storing information relating a state of the network connection and update this information in response to the network activity sensed by the firewall. The information stored in the database may include an idle time counter of the network connection. If the firewall determines that the network connection is active, it would reset this counter.

[0015]   The aforementioned network connection can be between a client and a server. In this case the probe may include a packet containing data from the server, the receipt of which has been already acknowledged by the client. The network activity may include a response from

5

the client indicating a condition of the network connection. Specifically, the response of the client may include a data receipt acknowledgment if the network connection is active and an error message if the network connection is not active. The probe can be nondestructive with respect to the network connection and it can be generated by the firewall. Alternative implementations of the probe are possible.

DESCRIPTION OF THE DRAWINGS

[0016]   Various embodiments of the present invention will now be described in detail by way of example only, and not by way of limitation, with reference to the attached drawings wherein identical or similar elements are designated with like numerals.

[0017]   FIG. 1 illustrates operation of a conventional firewall;

[0018]   FIG. 2 illustrates a typical network architecture utilizing a firewall;

[0019]   FIG. 3 illustrates operation of one embodiment of the inventive firewall.

DETAILED DESCRIPTION

[0020]   To overcome the above limitations and disadvantages attributable to the conventional firewall architecture, the inventive firewall automatically identifies active persistent network connections and keeps these connections alive.

[0021]   A typical secure network configuration using a firewall is illustrated in Fig. 2. Secure private network 7 links hosts 1, 2, and 3 together. This network is connected to the external global network 5, such as Internet, using a secure firewall computer 4. This computer

enforces security policy of the private network by filtering out network packets of connections that are in violation of this security policy. On the other hand, the connections complying with the security policy are being permitted by the firewall 4. For example, traveling employee may telnet into computer 2, located on the private network 7 from a remote host 6, connected to the Internet 5, assuming that the security policy of the private network 7 allows such a connection. This connection may become idle after a period of time.

[0022] According to an embodiment of the inventive method illustrated in Fig. 3., when the firewall 4 determines at 20 that a particular network connection has been idle for a predetermined period of time, the inventive firewall 4 does not automatically delete the connection's state from its database. Instead, the inventive firewall 4 tries to find out if the connection is an active persistent connection which should be kept alive for a longer period of time. To this end, the inventive stateful firewall sends out a message or a probe at 21 before deleting the state information entry of an idle connection from its database. The inventive probe is designed to elicit responses from the participants of the network connection that would provide information on the current condition of the network connection. The firewall then senses the network activity caused by these responses at 22 and determines if the connection in question is still active and should be kept alive, see Fig. 3 at 23. If the network connection is determined to be active, the corresponding idle time counter in the firewall database is reset at 24. Otherwise, the connection state entry is deleted from the database at 25. The operation of the algorithm terminates at 26. If the connection is determined by the firewall not to be idle, the firewall does not alter its state in the database.

[0023] In one embodiment of the invention, the aforementioned probe sent by the

firewall is designed to be nondestructive to the network connection. The probe elicits a network activity either by the server or by the client participating in the connection. The term "network activity" will be used herein to refer to generating a network message or packet or exchanging messages or packets in accordance with a network protocol. If the firewall then determines that this activity characterizes an active network connection, it would reset the idle time counter used by the firewall to identify the idle connections. This, in turn, would prevent the firewall from deleting the state of the corresponding persistent network connection.

[0024] The specific probe used in one embodiment of the invention is known as a BSD4.3 keepalive probe. This probe applies to TCP/IP connections. Specifically, the probe comprises a fake TCP/IP data packet sending the client data from the server. The data sent to the client is the data that the client has already acknowledged receiving. The following is an exemplary embodiment of such a probe.

Server sends: "Here is the data at position 100"

Client sends: "I got the data at position 100"

—idle—

Probe: "Here is the data at position 99"

Client sends: "I already acknowledged getting the data up to position 100".

As will be appreciated by those of skill in the art, the exemplary probe is arranged such that it comprises a copy of a message and/or data that have already been sent to the client by the server during preceding client–server communication. Accordingly, the client has already acknowledged receiving these data and, therefore, the client responds with the message "I already acknowledged getting the data up to position 100."

[0025]   The firewall passes the client's reply to the server who ignores the probe packet and the client's response. The firewall monitors the above client–server communication and determines that the network connection is still active. Accordingly, the firewall resets its idle time counter and keeps the connection alive.

[0026]   In the event the client has deleted the connection, upon the receipt of the probe packet the client will respond with the error message indicating that the collection is not active, followed by a RESET instruction. The firewall will sense this information and delete the corresponding connection state entry.

[0027]   In the event the server has deleted the connection, the server will respond with the message indicating that it never sent data at position 100 followed by a RESET. This will cause the firewall and the client to destroy the connection.

[0028]   Finally, it will be appreciated by those of skill in the art that if the client host is down, no responses are ever elicited and the inventive firewall will expire the connection as the conventional one.

[0029]   While the invention has been described herein with reference to preferred embodiments thereof, it will be readily apparent to persons of skill in the art that various modifications in form of detail can be made with respect thereto without departing from the spirit and scope of the invention as defined in and by the appended claims. For example, the present invention is not limited to TCP/IP connections. The inventive concept of identifying active persistent network connections before deleting them can apply to other network architectures based on a wide variety of network communication protocols. The specific format and content of the probe sent by the firewall is also not critical to the invention. The probe can

be implemented in a variety of formats and need only to elicit responses from the participants of the network connection. Finally, it is not essential that the probe be sent by the firewall. Any other participant of the network connection or any additional network entity can generate and send the probe.

[0030] Those of skill in the art will undoubtedly appreciate that the invention can be implemented on a vide variety of computer systems including, but not limited to, general purpose computers and special purpose computers such as network appliances. As well known in the art, a computer consists at least of a central processing unit, a memory unit, and an input/output interface. The aforementioned computer components can be arranged separately, or they can be combined together into a single unit. The computer memory unit may include a random access memory (RAM) and/or read only memory (ROM). The present invention can be implemented as a computer program embodied in any tangible storage medium, or loaded into the computer memory by any known means. As an alternative to implementing the present invention as a computer program, the present invention can be also embodied into an electronic circuit. This embodiment may provide an improved performance characteristics.